

AD684208
MTP-319

A MAXIMUM UTILITY SOLUTION TO A VEHICLE CONSTRAINED TANKER SCHEDULING PROBLEM

M. BELLMORE

S. LUBORE

G. BENNINGTON

AUGUST 1968



THE MITRE CORPORATION

Reproduced by the
CLEARINGHOUSE
for Federal Scientific & Technical
Information Springfield Va. 22151

This document has been approved for public
release and sale; its distribution is unlimited.

A MAXIMUM UTILITY SOLUTION TO A VEHICLE CONSTRAINED TANKER SCHEDULING PROBLEM

M. BELLMORE
JOHNS HOPKINS UNIVERSITY
G. BENNINGTON
S. LUBORE
THE MITRE CORPORATION

AUGUST 1968



ABSTRACT

A modification to the Dantzig and Fulkerson tanker scheduling problem is described. An insufficient number of vehicles and a utility associated with each vehicle delivery are assumed. The new problem is shown to be equivalent to a transshipment problem, the solution of which is the same as the maximal utility solution of the modified tanker scheduling problem. An example is given.

TABLE OF CONTENTS

	<u>Page</u>
List of Illustrations	ii
SECTION I INTRODUCTION	1
SECTION II NOTATION	2
SECTION III EXAMPLE	4
SECTION IV DETERMINATION OF ARC COSTS AND PROBLEM SOLUTION	10
Theorem	12
Proof	12
Method	14
Numerical Results for the Previous Example	14
REFERENCES	17

LIST OF ILLUSTRATIONS

	<u>Page</u>
Figure 1 Construction of Delivery Arcs	6
Figure 2 Addition of Reassignment Arcs	7
Figure 3 Completed Graph	9
Figure 4 Determination of Arc Costs	11

SECTION I

INTRODUCTION

This paper treats a modification of the tanker scheduling problem solved by Dantzig and Fulkerson.^[1] The problem was to minimize the number of vehicles, M' , needed to meet the entire schedule with zero deviation (i.e., no shipment is to be early or late). Dantzig and Fulkerson were able to show that what appeared to be a combinatorial problem could actually be formulated as a Hitchcock transportation problem.

The problem considered in this paper treats the case when there are insufficient vehicles, $M < M'$,* to meet the entire schedule, and, therefore, some deliveries must be cancelled. Utilities are assumed for each delivery, and the schedule is found which maximizes the sum of the utilities of the deliveries made with zero deviations.

A directed linear graph, G' , is constructed with the property that each chain is a feasible vehicle schedule. A set of costs is determined by employing a longest chain algorithm. A new problem, in the form of the transshipment problem, is solved to obtain the maximum flow solution with minimum cost. This flow of value M is shown to represent a feasible scheduling of M vehicles, which maximizes the total utility of deliveries.

If all utilities are positive and equal, then the number of shipments made with zero deviation is maximized. For simplicity, it is assumed that the utilities are positive integers.

(Wherever positive integers are assumed in this paper, any set of positive rational numbers might also be assumed.)

*The method can be used also when $M = M'$.

SECTION II

NOTATION

Throughout this paper, the notation of Dantzig and Fulkerson^[1] will be adhered to wherever possible. When needed, additional notation in the spirit of Ford and Fulkerson^[2] will be employed.

Let $I = \{i\}$ correspond to the set of ports at which shipments can originate; let $J = \{j\}$ correspond to the set of ports at which a delivery is to be made.* Associated with the sets I and J are sets of positive integers $A = \{a_{ij}\}$, the transit time from i to j (including loading and unloading), and $B = \{b_{ij}\}$, the return travel time from j to i .

The k^{th} shipment from i to j requires $m_{ij}^k > 0$ vehicles, leaving (more accurately, starting to load at) port i at time t_{ij}^k and arriving (more accurately, completely unloaded) at port j at time

$$T_{ij}^k = t_{ij}^k + a_{ij} \quad (1)$$

Each partial shipment has a utility, $d_{ij}^k v_{ij}^k \leq m_{ij}^k v_{ij}^k$, where v_{ij}^k is the utility of a single vehicle delivery and d_{ij}^k are the number of vehicle loads delivered. It is assumed that $t_{ij}^k \leq t_{ij}^{k+1}$.

A directed graph G' , is defined consisting of sets of nodes, s' , s , X , Y , t . Each node $x \in X$ is denoted by a unique pair of positive integers (α, i) . A node $x \sim (\alpha, i)$ exists if, and only if, at least one $t_{ij}^k = \alpha$. In like manner, each node $y \in Y$ is denoted by a unique pair of positive integers (β, j) . A node $y \sim (\beta, j)$ exists if, and only if, at least one

*Note that the intersection of sets I and J is not necessarily null.

$T_{ij}^k = t_{ij}^k + a_{ij} = \beta$. Further, a source node, s' , corresponding to an artificial source of vehicles, a sink node, t , and a node s are defined.

For each pair of nodes, $x \sim (\alpha, i)$ and $y \sim (\beta, j)$, a set of P parallel arcs, $(x, y)_p$, is defined if, and only if, $t_{ij}^k = t_{ij}^{k+1} = \dots = t_{ij}^{k+P-1} = \alpha$. (Note also, that $T_{ij}^k = T_{ij}^{k+1} = \dots = T_{ij}^{k+P-1} = \beta$, since $T_{ij}^{k+p} = t_{ij}^{k+p} + a_{ij}$.) The capacity and utility of each arc are defined as $c_p(x, y) = m_{ij}^{k+p-1}$ and $v_p(x, y) = v_{ij}^{k+p-1}$ respectively, $p = 1, 2, \dots, P$.

Further, for each pair of nodes, $y \sim (\beta, j)$ and $x \sim (\alpha, i)$, an arc, $(y, x)_1$, is defined if, and only if, $\alpha \geq \beta + b_{ij}$; with capacity, $c_1(y, x) = \infty$, and utility, $v_1(y, x) = 0$. Finally, arcs, $(s, x)_1$, and arcs, $(y, t)_1$, are defined for $x \in X$ and $y \in Y$, with $c_1(s, x) = c_1(y, t) = \infty$ and $v_1(s, x) = v_1(y, t) = 0$.

SECTION III

EXAMPLE

Consider a tanker scheduling problem involving two origins and two destinations. The shipment departure times are given in a rectangular array, one row for each origin, and one column for each destination. In each space, (i, j) , is the sequence of shipment departure times:

	$j = 1$	$j = 2$	
$i = 1$	1	9, 9	$= \left\{ t_{ij}^k \right\}$
$i = 2$	9	1, 5	

The number of vehicle deliveries comprising each shipment is given below:

	$j = 1$	$j = 2$	
$i = 1$	2	1, 1	$= \left\{ m_{ij}^k \right\}$
$i = 2$	1	1, 1	

The utility of each vehicle delivery is:

	$j = 1$	$j = 2$	
$i = 1$	2	2, 1	$= \left\{ v_{ij}^k \right\}$
$i = 2$	2	2, 1	

For this example, assume that the transit times are:

	$j = 1$	$j = 2$	
$i = 1$	7	3	$= \left\{ a_{ij} \right\} = \left\{ b_{ij} \right\}$
$i = 2$	1	2	

From Equation (1) we construct the table of shipment delivery times and, thus, the time when vehicles are available for reassignment:

	$j = 1$	$j = 2$	
$i = 1$	8	12, 12	$= \left\{ T_{ij}^k \right\} = \left\{ t_{ij}^k + a_{ij} \right\}$
$i = 2$	10	3, 7	

Consider the shipment from port $i = 1$, departing at time $t_{11}^1 = 1$, consisting of $m_{11}^1 = 2$ vehicle loads, and arriving at port $j = 1$ at $T_{11}^1 = 8$. Each load has utility $v_{11}^1 = 2$, hence the total utility of this shipment is 4. For this delivery, construct the nodes, $x \sim (\alpha = 1, i = 1)$ and $y \sim (\beta = 8, j = 1)$, and the arc, $(x, y)_1$, with capacity, $c_1(x, y) = m_{11}^1 = 2$, and utility, $v_1(x, y) = v_{11}^1 = 2$. In a similar manner, the arcs and nodes corresponding to the other shipments are constructed as shown in Figure 1. (Note the two parallel arcs between $x \sim (\alpha = 9, i = 1)$ and $y \sim (\beta = 12, j = 2)$, which correspond to two vehicle loads having different utilities.)

Arcs are now added to correspond to feasible reassessments of vehicles from a destination port to an originating port. For example, consider the node, $y \sim (\beta = 3, j = 2)$. If this shipment is made, then the vehicle could be reassigned to any origin, $x \sim (\alpha, i)$, such that $\alpha \geq \beta + b_{ij}$, specifically, to either $y \sim (\alpha = 5, i = 2)$, $(\alpha = 9, i = 2)$, or $(\alpha = 9, i = 1)$. No other origins are feasible with respect to transit times. Each of these arcs has infinite capacity (recall arcs terminating at $y \sim (\beta, j)$ have finite capacity) and zero utility. Figure 2 illustrates the construction so far.

Since any shipment may be made by assigning a new vehicle instead of reassigning an existing vehicle, arcs are added from s to each node

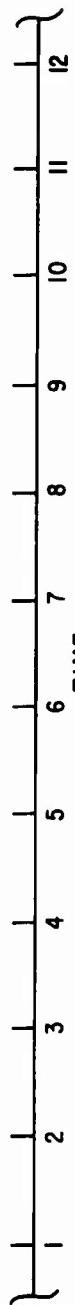
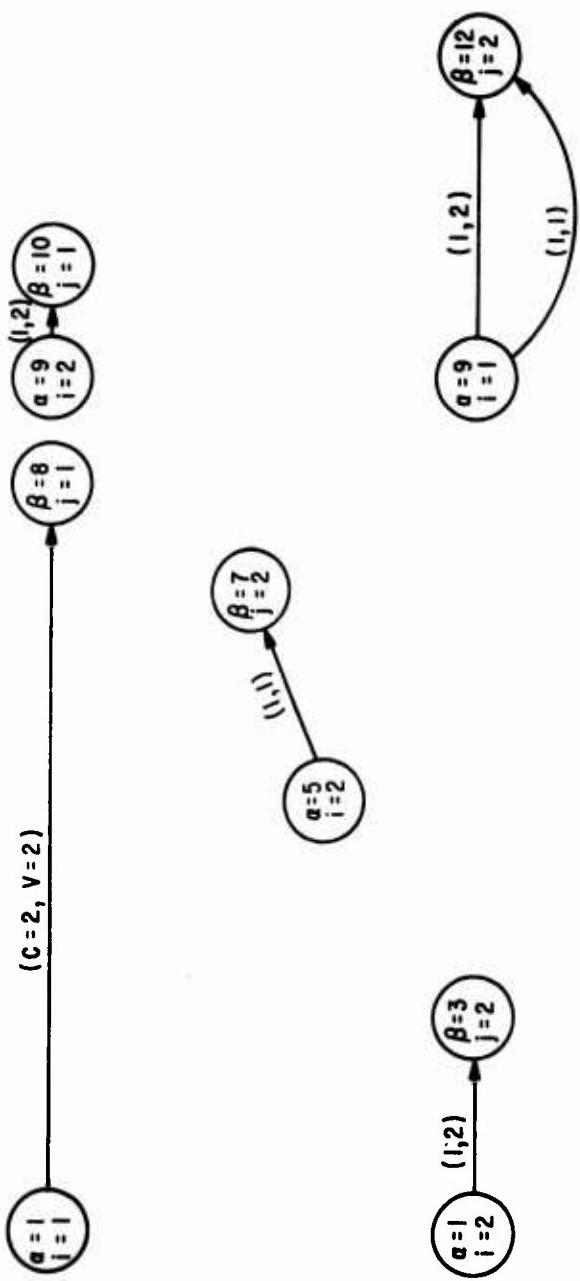
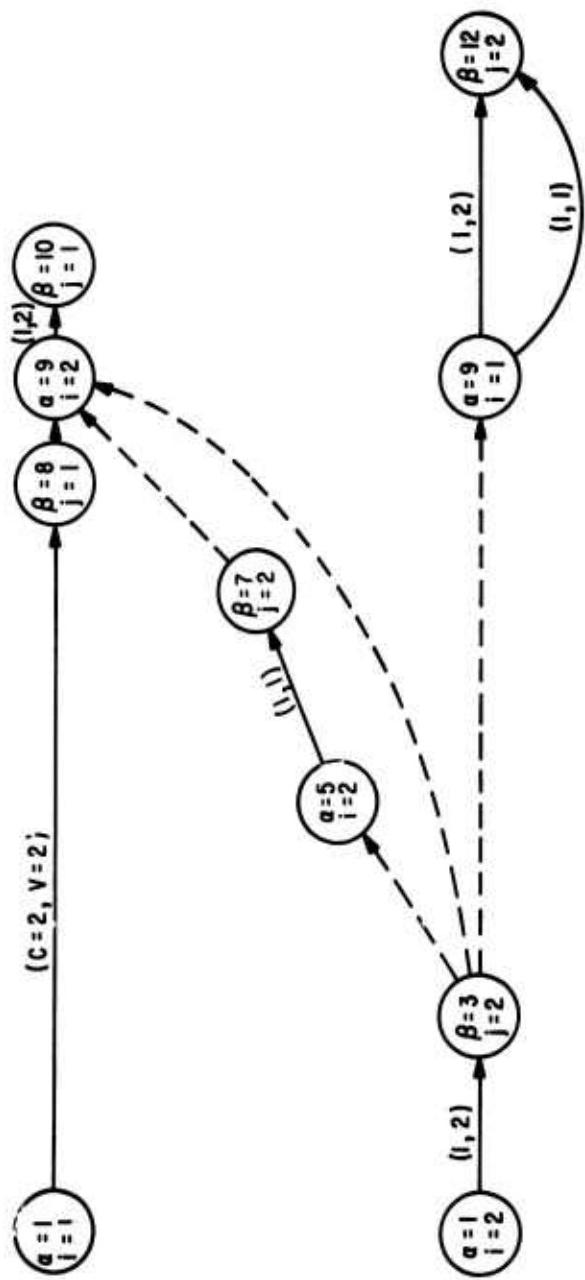


FIGURE 1
CONSTRUCTION OF DELIVERY ARCS



NOTE: ALL ARCS NOT DESIGNATED (C,V)
HAVE C = ∞ AND V = 0

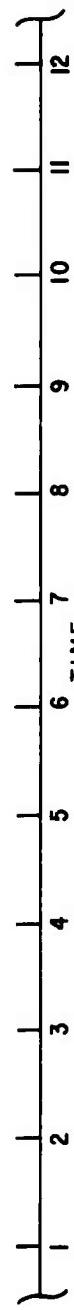


FIGURE 2
ADDITION OF REASSIGNMENT ARCS

$x \sim (\alpha, i)$. Each such arc has infinite capacity and zero utility. In a similar manner, we add arcs of infinite capacity and zero utility from each node, $y \sim (\beta, j)$, to t which correspond to not reassigning the vehicle after a delivery is made. Finally, we add node, s' , and arc, $(s', s)_1$, with capacity equal to the number of vehicles available, M , and zero utility.

The completed graph is illustrated in Figure 3.

From the method of construction, it is seen that there is a one-to-one correspondence between shipping schedules for M vehicles and flows of value M from s' to t in the constructed graph, G' .

G' is an acyclic graph, since arcs from $x \sim (\alpha, i)$ to $y \sim (\beta, j)$ have the property that $\beta > \alpha$ and arcs from $y \sim (\beta, j)$ to $x \sim (\alpha, i)$ have the property that $\alpha > \beta$, and, hence, for any chain, the times, α or β , are monotonically increasing and no cycle can occur.

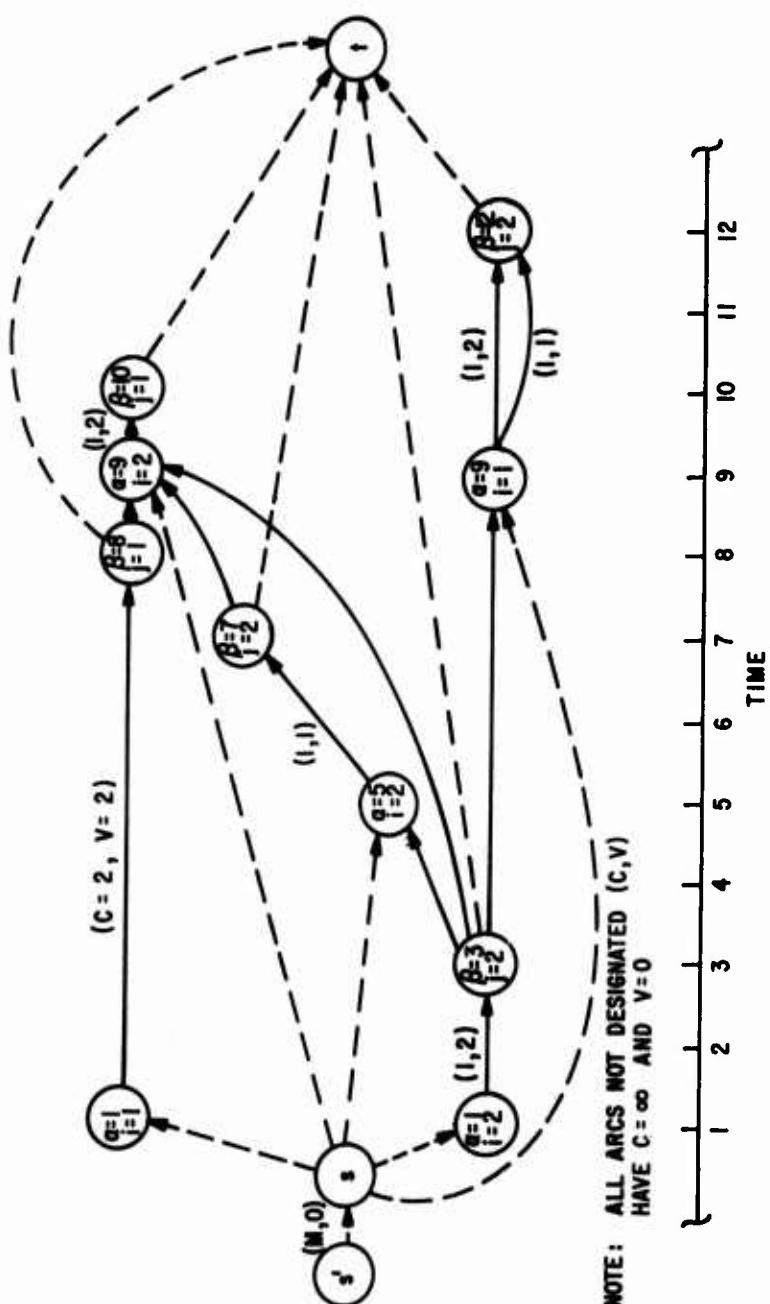


FIGURE 3
COMPLETED GRAPH

SECTION IV

DETERMINATION OF ARC COSTS AND PROBLEM SOLUTION

Since it has been assumed that M , the number of vehicles available, is less than or equal to M' , the minimum number of vehicles required to meet all shipping dates, a method to find a flow of value M which maximizes the total utility of all shipments delivered, i.e., use all available vehicles, is desired.

Next, a method is presented of assigning positive arc costs such that the minimum cost flow of value M corresponds to the maximum utility solution to the original problem.*

A simple modification of the shortest chain algorithm can be made which will find the longest chain from any node to a specific node in an acyclic graph. [3] [4] Denote $\pi(e)$ as the length of the longest chain from any node, e , in G' to node t , where the length of an arc is $v_p(e, f)$. At the termination of the longest chain algorithm, the $\pi(e)$ correspond to the node numbers.

Define $a_p(e, f) = \pi(e) - \pi(f) - v_p(e, f)$ as the cost associated with all arcs $(e, f)_p$ of G' . Figure 4 shows the costs and node numbers associated with the graph G' . Each node has been assigned an identifying number for future reference.

*Since G' is acyclic, an alternative solution method is to assign arc costs equal to the negative of the utility of the shipments over each arc and to find a minimum cost flow in G' . One could use the "Out-of Kilter" algorithm described in (2) or the primal method described in (5).

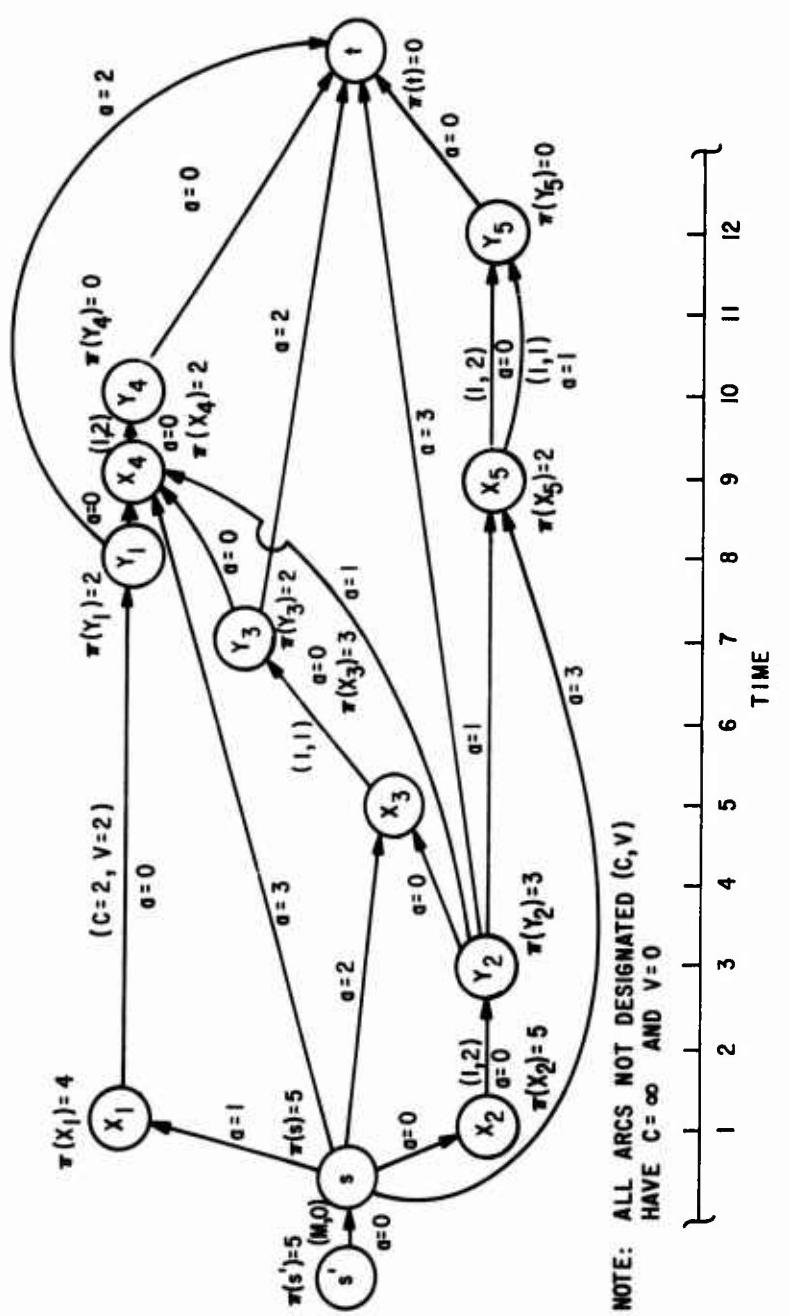


FIGURE 4
DETERMINATION OF ARC COSTS

THEOREM

The total cost associated with a unit chain flow, in the q^{th} chain,
 $[s', (s', s), s, (s, x_1)_1, x_1, (x_1, y_1)_{p_1}, y_1, \dots, x_n, (x_n, y_n)_{p_n}, y_n, (y_n, t)_1, t]$, in the directed graph G' , is given by:

$$C_q = \pi(s') - \sum_{(x_q, y_q)} v_{p_q}(x_q, y_q)$$

for all nodes $x_q \in X, y_q \in Y$ in the chain.

PROOF

The total chain cost is:

$$\begin{aligned} C_q &= a_1(s' \dots s) - a_1(s, x_1) + a_{p_1}(x_1, y_1) + a_1(y_1, x_2) \\ &\quad + \dots + a_1(y_n, t) \end{aligned}$$

$$\begin{aligned} C_q &= \pi(s') - \pi(s) - v_1(s', s) + \pi(s) - \pi(x_1) - v_1(s, x_1) \\ &\quad + \pi(x_1) - \pi(y_1) - v_{p_1}(x_1, y_1) + \pi(y_1) - \pi(x_2) \\ &\quad - v_1(y_1, x_2) + \dots + \pi(y_n) - \pi(t) - v_1(y_n, t) \end{aligned}$$

but, the only nonvanishing utilities are $v_{p_m}(x_m, y_m)$ and $\pi(t) = 0$.

Hence, letting $\sum_{(x_q, y_q)}$ be the summation over all arcs in the q^{th} chain
of G' ,

$$C_q = \pi(s') - \sum_{(x_q, y_q)} v_{p_q}(x_q, y_q) . \quad \underline{\text{Q. E. D.}}$$

Note in the theorem, the total cost of a unit flow in chain q is $C_q = \pi(s')$ - (the utility of loads delivered over chain q). The total cost of a chain flow of value M is the sum of the costs of the M unit chain flows, and is given by:

$$\bar{C} = \sum_{q=1}^M C_q = M \pi(s') - \sum_{q=1}^M \sum_{(x_q, y_q)} v_{p_q}(x_q, y_q)$$

$$\bar{C} = M \pi(s') - \sum_i \sum_j \sum_k d_{ij}^k v_{ij}^k = M\pi(s') - V$$

where,

$$V = \sum_i \sum_j \sum_k d_{ij}^k v_{ij}^k .$$

For M fixed, $M\pi(s')$ is a constant and the flow of value M which minimizes the total cost also maximizes the total utility of the loads delivered by M vehicles.

The transshipment problem (general minimum cost flow problem) is to find a maximum flow solution having the minimum cost. Since the maximum flow for our problem is $M = c_1(s', s) \leq M'$, the transshipment problem is to find a maximum flow of value M having minimum cost, and therefore maximum utility.*

* If it were not for $c_1(s', s)$, a flow of value M' or greater would be feasible.

The above construction of the graph, G' , has transformed the tanker scheduling problem with a limited number of vehicles into a transshipment problem. Most transshipment algorithms produce the optimum flow in node-arc form; however, since G' is an acyclic graph, the optimum flow can easily be transformed into arc-chain form. [2]

METHOD OF SOLUTION

- (1) Construct the directed graph, G' .
- (2) Use a longest chain algorithm to find $\pi(e)$, for all nodes, e .
- (3) Calculate the arc costs, $a_p(e, f) = \pi(e) - \pi(f) - v_p(e, f)$.
- (4) Solve this transshipment problem using an appropriate algorithm to obtain the maximum utility solution.
- (5) Construct individual vehicle schedules from the optimum arc flows in G' by transforming them into M unit chain flows, each corresponding to an optimum vehicle schedule.

NUMERICAL RESULTS FOR THE PREVIOUS EXAMPLE

For $M = 1$, the transshipment algorithm terminates with unit flow in the following arcs: $(s', s)_1, (s, x_2)_1, (x_2, y_2)_1, (y_2, x_3)_1, (x_3, y_3)_1, (y_3, x_4)_1, (x_4, y_4)_1, (y_4, t)_1$. The minimum cost is zero and the utility of deliveries made is $\pi(s') - 0 = 5$. Transformation of this flow yields the following schedule for the single vehicle (which is unique).

<u>Leave</u>	<u>From</u>	<u>Arrive</u>	<u>At</u>	<u>Utility</u>
$\alpha = 1$	i = 2	$\beta = 3$	j = 2	2
$\alpha = 5$	i = 2	$\beta = 7$	j = 2	1
$\alpha = 9$	i = 2	$\beta = 10$	j = 1	$\frac{2}{V = 5}$

The solution for $M = 2$ vehicles is:

Vehicle 1

<u>Leave</u>	<u>From</u>	<u>Arrive</u>	<u>At</u>	<u>Utility</u>
$\alpha = 1$	i = 1	$\beta = 8$	j = 1	2
$\alpha = 9$	i = 2	$\beta = 10$	j = 1	$\frac{2}{V_1 = 4}$

Vehicle 2

<u>Leave</u>	<u>From</u>	<u>Arrive</u>	<u>At</u>	<u>Utility</u>
$\alpha = 1$	i = 2	$\beta = 3$	j = 2	2
$\alpha = 9$	i = 1	$\beta = 12$	j = 2	$\frac{2}{V_2 = 4}$

$$V = V_1 + V_2 = 8$$

For two vehicles, the optimum schedule is unique and does not include the optimum schedule for the case when $M = 1$, which serves to show that a universally optimum schedule does not exist.

All deliveries can be made when $M = M' = 5$. If $M = 6$, the algorithm would use all six vehicles and should not be used for values of $M > M'$. M' can be determined either from the Dantzig and Fulkerson [1] algorithm or by using this algorithm iteratively, $M = 1, 2, \dots, M'$, and determining M' as the minimum value M for which all shipments

are made (i.e., $\bar{C} = M \pi(s') - \sum_i \sum_j \sum_k m_{ij}^k v_{ij}^k$) . Obviously, this

would not be done unless a complete parametric analysis is desired.

An alternative procedure, employing the "Out-of-Kilter" algorithm, would assign a small positive cost $v_1(s', s) = \epsilon > 0$ to the arc $(s', s)_1$; assign costs $v'_{pq}(x_q, y_q) = -v_{pq}(x_q, y_q)$ to the delivery arcs; and

determine the minimum cost flow solution in the graph G' . This procedure may be employed even when $M > M'$.

REFERENCES

1. G. B. Dantzig and D. R. Fulkerson, "Minimizing the Number of Tankers to Meet a Fixed Schedule," Naval Logistics Research Quarterly, Vol. 1, pp. 217-222, 1954.
2. L. R. Ford and D. R. Fulkerson, Flows in Networks, Princeton University Press, 1962.
3. D.R. Fulkerson, "A Network Flow Computation for Project Cost Curves," Management Science, Vol. 7, pp. 167-178, 1961.
4. J. E. Kelley, Jr., "Critical Path Planning and Scheduling: Mathematical Basis," Operations Research, Vol. 9, pp. 296-320, 1961.
5. Morton Klein, "A Primal Method for Minimum Cost Flows with Applications to the Assignment and Transportation Problems," Management Science, Vol. 14, pp. 205-220, 1967.

Security Classification**DOCUMENT CONTROL DATA - R & D**

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) The MITRE Corporation		2a. REPORT SECURITY CLASSIFICATION N/A
		2b. GROUP N/A
3. REPORT TITLE A Maximum Utility Solution to a Vehicle Constrained Tanker Scheduling Problem		
4. DESCRIPTIVE NOTES (Type of report and inclusive dates)		
5. AUTHOR(S) (First name, middle initial, last name) M. Bellmore G. Bennington S. Lubore		
6. REPORT DATE August 1968	7a. TOTAL NO. OF PAGES 21	7b. NO. OF REFS 5
8a. CONTRACT OR GRANT NO. AF 19(628)5165	8c. ORIGINATOR'S REPORT NUMBER(S) MTP 319	
8b. PROJECT NO. 31.6.0	8d. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)	
c.		
d.		
10. DISTRIBUTION STATEMENT This document has been approved for public release and sale; its distribution is unlimited.		
11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY Defense Communications Agency	
13. ABSTRACT A modification to the Dantzig and Fulkerson tanker scheduling problem is described. An insufficient number of vehicles and a utility associated with each vehicle delivery are assumed. The new problem is shown to be equivalent to a transshipment problem, the solution of which is the same as the maximal utility solution of the modified tanker scheduling problem. An example is given.		

DD FORM NOV. 1473

Security Classification

Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Linear Programming						
Network Theory						
Transportation						
Tanker Scheduling						
Scheduling						